Package 'gameofthrones'

November 26, 2026

Type Package

Title Palettes Inspired in the TV Show ``Game of Thrones''

Version 1.0.3

Maintainer Alejandro Jimenez Rico <aljrico@gmail.com>

Description Implementation of the characteristic palettes from the TV show ``Game of Thrones''.

License MIT + file LICENSE

Encoding UTF-8

LazyData TRUE

Depends R (>= 3.5.0)

Suggests hexbin (>= 1.27.0)

URL https://github.com/aljrico/gameofthrones

BugReports https://github.com/aljrico/gameofthrones/issues

Imports ggplot2 (>= 1.0.1)

RoxygenNote 7.3.3

NeedsCompilation no

Author Alejandro Jimenez Rico [aut, cre]

Repository CRAN

Date/Publication 2026-11-26 09:50:11 UTC

Contents

got 2
got.map 4
got_palettes 4
scale_got 5

Index **8**

got *Game of Thrones Colour Map.*

Description

This function creates a vector of n equally spaced colors along the 'got colour map'.

Usage

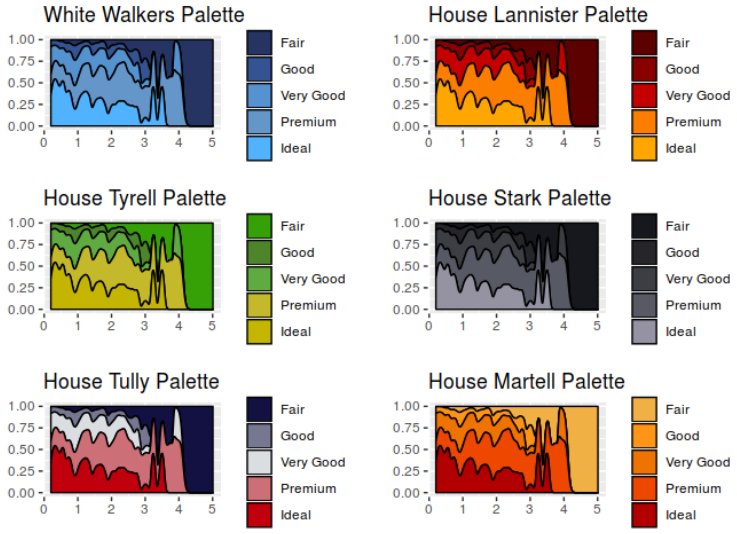
```
got(n, alpha = 1, begin = 0, end = 1, direction = 1, option = "Daenerys")

got_pal(alpha = 1, begin = 0, end = 1, direction = 1, option = "Daenerys")

gameofthrones(
  n,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  option = "Daenerys"
)
```

Arguments

n	The number of colors (≥ 1) to be in the palette.
alpha	The alpha transparency, a number in [0,1], see argument alpha in hsv.
begin	The (corrected) hue in [0,1] at which the got colormap begins.
end	The (corrected) hue in [0,1] at which the got colormap ends.
direction	Sets the order of colors in the scale. If 1, the default, colors are ordered from darkest to lightest. If -1, the order of colors is reversed.
option	A character string indicating the colourmap to use. It is not case-sensible

Details

Here are the color scales:



Semi-transparent colors ($0 < alpha < 1$) are supported only on some devices: see `rgb`.

Value

`got` returns a character vector, `cv`, of color hex codes. This can be used either to create a user-defined color palette for subsequent graphics by `palette(cv)`, a `col =` specification in graphics functions or in `par`.

Author(s)

Alejandro Jiménez Rico <aljrico@gmail.com>, Personal Blog

Examples

```
library(ggplot2)
library(hexbin)

dat <- data.frame(x = rnorm(1e4), y = rnorm(1e4))

ggplot(dat, aes(x = x, y = y)) +
  geom_hex() + coord_fixed() +
  scale_fill_gradientn(colours = got(128, option = "targaryen")) +
  theme_minimal()

pal <- got(256, option = "Wildfire")
image(volcano, col = pal)
```

got.map *'Game of Thrones' colour map*

Description

A dataset containing the colour palettes from the TV show 'Game of Thrones'.

Usage

```
got.map
```

Format

A data frame containing all the colours used in the palette:

- V1: Red value
- V2: Green value
- V3: Blue value
- option: Refers to the houses of Westeros, or other colourmaps. It is intended to be a general option for choosing the specific colour palette.

got_palettes *Complete list of palettes*

Description

Use got to construct palettes of desired length.

Usage

```
got_palettes
```

Format

An object of class list of length 18.

scale_got *Game of Thrones colour scales*

Description

Uses the Game of Thrones color scale.

Usage

```
scale_colour_got(
  ...,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  discrete = FALSE,
  option = "Daenerys"
)

scale_color_got(
  ...,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  discrete = FALSE,
  option = "Daenerys"
)

scale_fill_got(
  ...,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  discrete = FALSE,
  option = "Daenerys"
)

scale_colour_got_d(
  ...,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  option = "Daenerys",
  aesthetics = "colour"
```

```
)

scale_color_got_d(
  ...,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  option = "Daenerys",
  aesthetics = "colour"
)

scale_fill_got_d(
  ...,
  alpha = 1,
  begin = 0,
  end = 1,
  direction = 1,
  option = "Daenerys",
  aesthetics = "fill"
)
```

Arguments

...	parameters to discrete_scale or scale_fill_gradientn
alpha	pass through parameter to got
begin	The (corrected) hue in [0,1] at which the got colormap begins.
end	The (corrected) hue in [0,1] at which the got colormap ends.
direction	Sets the order of colors in the scale. If 1, the default, colors are as output by got_pal. If -1, the order of colors is reversed.
discrete	generate a discrete palette? (default: FALSE - generate continuous palette)
option	A character string indicating the colourmap to use.
aesthetics	Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply colour settings to the colour and fill aesthetics at the same time, via aesthetics = c("colour", "fill"). Parameter inherited from 'ggplot2'.

Details

For discrete == FALSE (the default) all other arguments are as to scale_fill_gradientn or scale_color_gradientn. Otherwise the function will return a discrete_scale with the plot-computed number of colors.

Value

A 'ggplot2' scale object (a ggproto object inheriting from Scale) that sets the colours of a plot when added to it with +. The _d functions, and the others when discrete = TRUE, return a discrete scale (class ScaleDiscrete) that takes one colour from the palette per data level. The others return a continuous scale (class ScaleContinuous) that maps numeric values onto a gradient of the palette.

Author(s)

Alejandro Jiménez Rico <aljrico@gmail.com>

Examples

```
library(ggplot2)

ggplot(mtcars, aes(factor(cyl), fill=factor(vs))) +
geom_bar() +
scale_fill_got(discrete = TRUE, option = "Daenerys")

ggplot(mtcars, aes(factor(gear), fill=factor(carb))) +
geom_bar() +
scale_fill_got(discrete = TRUE, option = "Tully")

ggplot(mtcars, aes(x = mpg, y = disp, colour = hp)) +
geom_point(size = 2) +
scale_colour_got(option = "Lannister")
```

Index

∗ **datasets**
got.map, 4
got_palettes, 4

gameofthrones *(*got*)*, 2
got, 2, *4*
got.map, 4
got_pal *(*got*)*, 2
got_palettes, 4

hsv, *2*

rgb, *3*

scale_color_got *(*scale_got*)*, 5
scale_color_got_d *(*scale_got*)*, 5
scale_color_gradientn, *6*
scale_colour_got *(*scale_got*)*, 5
scale_colour_got_d *(*scale_got*)*, 5
scale_fill_got *(*scale_got*)*, 5
scale_fill_got_d *(*scale_got*)*, 5
scale_fill_gradientn, *6*
scale_got, 5